## Introduction

**Representational State Transfer** is the abbreviation for REST. Rest assured is a **Java-based library** that is used for **testing RESTful web services**.

Rest assured is a tool that helps developers to test and verify the functionality of their RESTful web services.

It provides an easy-to-use syntax for writing automated tests that **simulate HTTP requests and responses**, allowing developers to check that their APIs are working as expected.

In other words, Rest assured is a library that helps to **automate the testing of web APIs**. In this article, we will discuss about Rest assured interview questions.

For more information, you can visit [this](https://www.codingninjas.com/studio/library/introduction-to-rest-assured) article and enhance your knowledge regarding REST Assured.

**Top Rest Assured Interview Questions for Freshers**

In this section, we have provided some fresher Level Questions with answers on Rest Assured that can be asked during the interview.

**1. What protocol is utilized by RESTful Web Services?**

RESTful web services typically utilize the HTTP (Hypertext Transfer Protocol) as the underlying protocol for communication. HTTP is the standard protocol for data transmission on the World Wide Web and provides a simple and lightweight way for clients to request and interact with resources on a web server.

**2. What is Rest Assured, and why is it used for API testing?**

Rest Assured is a popular Java-based library used for testing RESTful APIs. It provides a set of easy-to-use APIs for writing test cases for RESTful web services.

Rest Assured is used for API testing because it allows you to test the functionality of RESTful web services, ensuring that the API is working as expected.

By using Rest Assured, you can validate the response of the API and also check for the correct status codes, and validate the structure of the response.

**3. What are the advantages of using Rest Assured for API testing?**

Some of the advantages of using Rest Assured for API testing are:-

* **Easy to use**: Rest Assured is easy to use, even for testers who are new to API testing.
* **Supports multiple protocols**: Rest Assured supports a wide range of protocols, including HTTP, HTTPS, and OAuth.
* **Flexible and extensible**: Rest Assured is flexible and extensible. It allows testers to customize and extend its functionality as needed.
* **Integrates with popular testing frameworks**: Rest Assured integrates with popular testing frameworks, such as JUnit and TestNG.

**4. What are the different HTTP methods that Rest Assured supports?**

Rest Assured supports all the standard HTTP methods that are commonly used in RESTful API design.

These methods are:

* **GET**: This method is used to retrieve resource(s) from the server.
* **POST**: This method is used to create a new resource on the server.
* **PUT**: This method is used to update an existing resource on the server.
* **DELETE**: This method is used to delete a resource from the server.
* **PATCH**: This method is used to apply partial modifications to a resource.
* **TRACE**: This method is used to retrieve a diagnostic trace of the actions performed by a server.

**5. What is the difference between Rest Assured and other API testing tools?**

Rest Assured is a popular open-source Java library for testing RESTful APIs, and there are several other API testing tools available in the market.

Some of the key differences between Rest Assured and other API testing tools are:

* **Programming language**: Rest Assured is a Java library, so it requires knowledge of Java programming to use effectively. Other tools may use different programming languages, such as Python or JavaScript.
* **Syntax**: Rest Assured has a simple and intuitive syntax that is designed to make it easy to write and execute tests. Other tools may have a more complex syntax that requires a steeper learning curve.
* **Customization**: Rest Assured is highly customizable, and testers can extend its functionality or create custom matchers as needed. Other tools may not provide the same level of customization.
* **Integration**: Rest Assured integrates well with popular Java-based testing frameworks, such as JUnit and TestNG. Other tools may integrate with different testing frameworks or require a more complex setup.

**6. How do you send a GET request using Rest Assured?**

To send a GET request using Rest Assured, you can use the get() method provided by Rest Assured.

For example, the following code will send a GET request to the specified URL:

Response response = RestAssured.get("URL\_To\_Be\_Mentioned");

**7. How do you verify the response status code in Rest Assured?**

To verify the response status code in Rest Assured, you can use the statusCode() method provided by Rest Assured.

For example, the following code will verify that the response status code is 200:

given()

 .get("URL\_To\_Be\_Mentioned")

.then()

 .statusCode(200);

**8. How do you extract data from a response in Rest Assured?**

To extract data from a response in Rest Assured, you can use the extract() method provided by Rest Assured.

For example, the following code will extract the value of the "title" field from the response:

Response response = RestAssured.get("URL\_To\_Be\_Mentioned");

String title = response.jsonPath().get("title");

**9. What is the difference between given() and when() in Rest Assured?**

In Rest Assured, given() is used to set up the request, while when() is used to send the request. given() allows you to specify the request method, headers, cookies, and other details, while when() allows you to send the request and get the response.

For example, the following code sets up a GET request and sends it:

given()

 .header("Authorization", "Bearer myToken")

.when()

 .get("URL\_To\_Be\_Mentioned")

.then()

 .statusCode(200);

**10. What is the maximum size of a payload that can be sent via the POST method?**

The HTTP specification does not specify a maximum size for posts. They are typically constrained by the web server or the programming technology used to process form submissions.

**11. Describe REST.**

REST (Representational State Transfer) is a set of rules and guidelines that describe how web-based applications should communicate with each other. REST defines a standard way of making requests to a web server using HTTP, which is the protocol used by web browsers to communicate with servers.

RESTful web services make it easier for different software systems and devices to communicate with each other over the internet. These web services are based on a uniform interface, which includes HTTP verbs such as GET, POST, PUT, and DELETE.

The main advantage of using RESTful web services is that they are scalable and adaptable, making it easier to develop and maintain web applications. They are widely used in modern web development and can be used to build applications for different types of devices, such as computers, smartphones, and tablets.

**Top Rest Assured Interview Questions for Intermediate**

In this section, we have provided some intermediate Level Questions with answers on Rest Assured that can be asked during the interview.

**12. Can a GET query be made in place of a PUT to create a resource?**

No, it's not appropriate to use a GET query to create a resource. According to the RESTful principles, a GET request is meant to retrieve a resource, while a PUT request is used to update or create a resource at a specific URI.

Using a GET request to create a resource would violate this principle because a GET request is not meant to modify server-side data.

Additionally, GET requests are typically cached by web browsers and other intermediaries, so it's possible that the request could be repeated without the client's knowledge, which could result in unintended consequences and data inconsistencies.

**13.  Explain REST Assured method chaining.**

REST Assured is a Java-based library used for testing RESTful Web Services. One of the key features of REST Assured is its method-chaining capability, which allows for a more concise and readable syntax for constructing API requests and assertions.

In method chaining, each method call returns a reference to the current object, which can be used to call the next method in the chain. This can be especially useful when making API requests that involve multiple parameters or headers.

**14. Name the essential elements of an HTTP response.**

An HTTP response typically includes the following essential elements:

* **Status line:** The first line of an HTTP response, which includes the HTTP version, status code, and a short message describing the status.
* **Headers:** Additional metadata about the response, such as content type, cache-control directives, cookies, etc.
* **Body:**The content of the response may be empty or contain various types of data, such as HTML, JSON, XML, or plain text.

**15. What is jsonPath in Rest Assured?**

JSONPath is a syntax used to extract specific data from a JSON string. It is similar to XPath syntax for XML data. In REST Assured, JSONPath is used to extract data from JSON responses returned from API requests.

JSONPath expressions are used to traverse the JSON data and locate specific data elements. The expressions are composed of various operators that can be used to navigate the JSON structure and select the desired data.

**16. What technique does caching use?**

Caching is a technique used to improve the performance and reduce the network traffic of web applications by storing frequently accessed data in memory or on disk and serving it from the cache rather than fetching it from the original source every time it is requested.

1. **Browser caching:** The browser stores a copy of the data in its cache, which can be used to serve subsequent requests for the same data.
2. **Proxy caching:** A proxy server sits between the client and the server, and stores a copy of the data in its cache. When the client requests the data, the proxy can serve it from the cache instead of forwarding the request to the server.
3. **Content Delivery Network (CDN) caching:**A CDN stores a copy of the data in its cache, which can be served to clients from a location that is closer to them, reducing the response time and network traffic.
4. **Application caching:** The application stores a copy of the data in memory or on disk, which can be used to serve subsequent requests for the same data.

**17. What is Client-server architecture?**

Client-server architecture is a computing model in which the workload is divided between two types of processes: client processes and server processes. The client processes request services or resources from the server processes, which provide the requested services or resources.

In this architecture, the client is usually a user-facing interface that sends requests to the server, while the server is a back-end process that provides the requested services or resources to the client. The communication between the client and the server is typically done over a network, using a communication protocol such as TCP/IP.

**18. Describe JSON.**

JSON stands for JavaScript Object Notation. It is a lightweight data interchange format that is easy for humans to read and write and easy for machines to parse and generate. JSON is often used to transmit data between a server and a client in web applications and is commonly used as an alternative to XML.

JSON data is represented as a collection of key-value pairs, similar to objects in JavaScript. The key is a string, and the value can be a string, number, Boolean, array, or another JSON object. The syntax of JSON is simple and intuitive, making it easy to read and understand even for non-technical people.

Here is an example of a simple JSON object:

{

 "name": "John Doe",

 "age": 30,

 "email": "john.doe@example.com",

 "address": {

   "street": "123 Main St",

   "city": "Anytown",

   "state": "CA",

   "zip": "12345"

 },

 "hobbies": ["reading", "traveling", "cooking"]

}

**19. What procedures and techniques are used to verify the REST API response in Rest Assured?**

Below are mentioned some procedures and techniques which are used to verify the REST API response in Rest Assured:

1. **Status code validation:** Rest Assured provides an assertion method called statusCode() that allows you to verify the HTTP status code of the response. For example, you can use statusCode(200) to ensure that the response status code is 200 (OK).
2. **Header validation:** Rest Assured provides assertion methods for verifying specific headers in the response, such as header("Content-Type", "application/json") to check if the "Content-Type" header is set to "application/json".
3. **Body validation:** Rest Assured provides several assertion methods for validating the response body.
4. **Response time validation:** Rest Assured provides an assertion method called time() that allows you to verify the response time of the API call.
5. **Cookie validation:** Rest Assured provides assertion methods for validating cookies in the response, such as cookie("session\_id", "12345") to check if the "session\_id" cookie is set to "12345".
6. **Schema validation:** Rest Assured allows you to perform schema validation on the response using tools like JsonSchemaValidator or XmlSchemaValidator.

**20. Why does Rest Assured use static import?**

Rest Assured uses static imports to provide a more concise and readable syntax for writing test code. With static imports, you can call methods from the Rest Assured library without having to prefix them with the class name or instantiate an object.

For example, consider the following code snippet that sends a GET request to a REST API and validates the response status code:

import static io.restassured.RestAssured.\*;

import static org.hamcrest.Matchers.\*;

given().

 param("key", "value").

when().

 get("/api/endpoint").

then().

 statusCode(200).

 body("key", equalTo("value"));

With static imports, you can call the given(), when(), and then() methods from Rest Assured directly without having to prefix them with the class name RestAssured. You can also call the statusCode() and body() methods directly from the then() method without having to instantiate an object of the ValidatableResponse class.

**21. Define a resource in REST.**

In REST (Representational State Transfer), a resource is any information or entity that can be identified by a unique identifier, typically represented by a URL (Uniform Resource Locator). A resource can be anything that can be stored, retrieved, or manipulated by a client, such as a document, an image, a user profile, or any other piece of data.

A resource in REST is typically represented by a specific endpoint on a web server, and clients interact with that resource using standard HTTP methods such as GET, POST, PUT, PATCH, and DELETE.

**Top Rest Assured Interview Questions for Experienced**

In this section, we have provided some experienced level questions with answers on Rest Assured that can be asked during the interview.

**22. What are the best practices to follow when using Rest Assured for API testing?**

There are many best practices to follow when using Rest Assured for API testing. Some of them are:

* **Write clear and readable tests**: Use descriptive method names, organize your tests into logical groups, and use comments to explain what the test is doing.
* **Verify response codes and headers**: Verify that your API returns the correct response codes and headers. Use assertions to ensure that your API is returning the expected results.
* **Use JSON and XML parsers**: Use JSON and XML parsers to parse and verify responses. Rest Assured has built-in support for parsing JSON and XML.
* **Use logging and debugging**: Use logging and debugging to identify issues in your tests. Rest Assured provides logging features to help you track down issues.
* **Use environment-specific configuration**: Use environment-specific configuration to specify different settings for your API based on the environment you are testing in.
* **Use code reviews**: Conduct code reviews to ensure that your tests are well-structured, maintainable, and follow best practices.
* **Keep your tests up to date**: Keep your tests up to date as the API changes. This will help you catch issues early and ensure that your tests remain relevant.

**23. What is a RESTFul Web service's payload?**

A RESTful web service's payload is the data that is sent and received over the network in the request and response messages.

In the context of a RESTful web service, the payload typically consists of a representation of a resource, which can be in various formats such as JSON, XML, or plain text. The payload of a RESTful web service's request message typically includes information about the operation to be performed on the resource.

Overall, the payload of a RESTful web service is a critical component of the overall design, as it determines how the service communicates with its clients and the types of data that can be exchanged.

**24. How do you extract the values of JSON, and how do you validate response?**

In Rest Assured, you can extract the values of JSON from the response using the JsonPath library. The JsonPath library provides a simple syntax for navigating through JSON objects and extracting the values of specific fields.

To extract the values of JSON fields, you can use the JsonPath object returned by the getResponse() method.

Here's an example:

Response response = given()

   .get("/api/endpoint")

   .then()

   .extract()

   .response();

JsonPath jsonPath = response.jsonPath();

String name = jsonPath.get("name");

int age = jsonPath.get("age");

In this example, we are sending a GET request to /api/endpoint and extracting the response as a Response object. We then create a JsonPath object from the response using the response.jsonPath() method. Finally, we extract the values of the name and age fields using the get() method.

To validate the response, you can use Rest Assured's built-in assertion methods or Hamcrest matchers. Here's an example of using assertion methods to validate the response:

given()

   .get("/api/endpoint")

   .then()

   .statusCode(200)

   .body("name", equalTo("John Doe"))

   .body("age", greaterThan(18));

In this example, we are sending a GET request to /api/endpoint and validating the response using the statusCode() and body() assertion methods. The body() method allows you to specify the JSON path of the field you want to validate and the expected value. In this case, we are checking that the name field is equal to "John Doe" and that the age field is greater than 18.

**25. How many types of Authentication are there in POSTMAN/ Rest-Assured?**

In both POSTMAN and Rest-Assured, there are several types of authentication methods that can be used to authenticate a request to a REST API.

Here are some of the most common types of authentication:

1. **Basic Authentication:** This is the simplest form of authentication, which uses a username and password in the request header.
2. [**Digest Authentication**](https://www.codingninjas.com/studio/library/digest-auth-in-postman)**:** This type of authentication is similar to Basic Authentication, but it uses a more secure hashing algorithm to protect the credentials.
3. [**OAuth 1.0/2.0 Authentication**](https://www.codingninjas.com/studio/library/oauth-1-in-postman)**:**OAuth is a protocol for authorization, which allows users to grant access to their resources to third-party applications without sharing their passwords.
4. **API Key Authentication:** This method involves using a unique API key to authenticate the request.
5. **JWT Authentication:** JSON Web Tokens (JWT) are a type of token-based authentication that is commonly used in RESTful applications.

**26. What are the dependencies for Rest-Assured?**

To use Rest-Assured in a Java project, you need to include the following dependencies in our project:

1. **Rest-Assured:** This is the main dependency for Rest-Assured, which includes the core functionality for making HTTP requests and validating responses.
2. **JSON/XML parser:** Rest-Assured uses a JSON parser to parse JSON responses and an XML parser to parse XML responses. You can choose any JSON/XML parser you prefer. Some popular choices include Jackson, Gson, and XMLUnit.
3. **Test Framework:** Rest-Assured works well with most popular Java test frameworks such as JUnit, TestNG, and Spock. You can choose the one that best fits your needs.
4. **HTTP Client:** Rest-Assured uses an HTTP client to make HTTP requests. You can choose any HTTP client you prefer. Some popular choices include Apache HttpClient, OkHttp, and Jersey.

**27.  Why would a programmer use REST Assured to automate RESTful services instead of Postman?**

REST Assured and Postman each have their own strengths and weaknesses, and the choice between them depends on the programmer's requirements and preferences. REST Assured is a good option for programmers who are comfortable with coding and want to incorporate API testing into a CI pipeline, while Postman is a good option for team members who want a more visual, user-friendly tool for exploring and testing APIs.

**28. How is chaining carried out in REST Assured?**

Chaining in REST Assured refers to the technique of method chaining, which is a way of invoking multiple methods on a single object in a sequence. This technique is used extensively in REST Assured to write concise and readable tests for REST APIs.

In REST Assured, method chaining is carried out by returning the same object from each method call, allowing the subsequent method to be invoked on the same object.

**29. What are Serialization and Deserialization in Rest Assured?**

In REST Assured, serialization and deserialization refer to the process of converting Java objects to and from JSON format. When making API requests with REST Assured, it is common to send and receive JSON data in the request and response bodies.

**Serialization:**Serialization in REST Assured refers to the process of converting a Java object into a JSON string that can be sent in the request body. REST Assured provides several methods for serializing Java objects into JSON format, including the ObjectMapper class and the JsonPath class.

**Deserialization:**Deserialization in REST Assured refers to the process of converting a JSON string from the response body into a Java object. REST Assured provides several methods for deserializing JSON data into Java objects, including the ObjectMapper class, the JsonPath class, and the Response class.

**30. What is the best way to keep sensitive data out of the log in rest assured?**

When using REST Assured to test APIs, it is important to keep sensitive data, such as passwords and API keys, out of the log. The log may contain sensitive information, which could be a security risk if it falls into the wrong hands.

There are several ways to keep sensitive data out of the log in REST Assured:

1. **Use environment variables or system properties:** Store sensitive data in environment variables or system properties, and access them in the test code using the System.getenv() or System.getProperty() methods. This way, sensitive data is not stored in the code, and is not visible in the log.
2. **Use logback configuration:**Configure logback to exclude certain log statements or exclude certain fields in the log.
3. **Use request and response filters:** Request and response filters can be used to modify the request or response before it is logged.
4. **Use log masking:**Log masking can be used to mask sensitive data in the log.

**31. How does Rest Assured work internally?**

Rest Assured is a library that provides a set of easy-to-use methods for testing RESTful APIs. When a developer writes a test case using Rest Assured, the library uses an HTTP client library to send an HTTP request to the API endpoint and receive a response. It then uses a library to deserialize the response data (often in JSON format) into Java objects.

The developer can then use the provided methods to check various aspects of the response, such as the status code, response headers, and response body. Rest Assured uses a fluent API design that makes it easy to write test cases in a readable and natural language-like syntax.

Must Read [Web Developer Interview Questions](https://www.codingninjas.com/studio/library/web-developer-interview-questions)

**Frequently Asked Questions**

**How do I prepare for a Rest assured interview?**

You can prepare for a Rest assured interview by understanding REST concepts, learning Java basics, mastering Rest Assured syntax, and practicing API testing. Also, grasp response validation, authentication, error handling, and testing best practices.

**How do you explain the Rest assured framework in an interview?**

You can explain the Rest assured framework in an interview like this “Rest Assured is a Java library for automating REST API tests. It simplifies testing with a fluent syntax, supports request/response handling, and  is used for functional and integration testing.”

### Which tool is used for Rest assured?

The primary tool for Rest Assured is Rest Assured itself, a Java library. It integrates with Java test frameworks and is managed with tools like Maven or Gradle. IDEs like IntelliJ IDEA or Eclipse are commonly used for writing Rest Assured tests.